Installing, Testing, and Customizing a CVMFS Server

John De Stefano
Grid/Experiment Services, RACF, BNL

Introduction

In addition to gaining expertise, and to simply testing an existing set of technologies, one of the goals of this exercise was to determine how flexible the CVMFS server installation could be: OSG desires to use the software as part of its planned architecture to distribute software repositories, but to obfuscate the underlying technology and unify its pieces under the OASIS project banner. For example, on the CVMFS server, the default mount point for software storage and distribution is `/cvmfs`; OSG would like to change this to `/oasis`. This was suspected to be problematic, as no documentation existed to support such a mount rename or relocation, and software configuration knobs to this effect are sparse and also undocumented.

Infrastructure

No hardware had been allocated beforehand for this purpose. After waiting a few days for feedback on available servers, two virtual RHEL 6 nodes were requested from GCE: one for the CMVFS client and one for the server; both nodes were deployed and received within a few hours on the same day [1].

General Installation and Troubleshooting

The client installation went smoothly, after working through a few bugs with packages and keys, but the server did not: a CVMFS server RPM is available for RHEL 6, but it contains hidden dependencies for CVMFS RPMs that are currently only available for EL 5: RedirFS, and a CVMFS kernel filter that depends on RedirFS. RedirFS was installed from official source, which worked, but compiling and installing the CVMFS RedirFS kernel filter failed due to a parameter compatibility issue. The RedirFS and kernel filter modules were then both built from the CVMFS source, but the kernel filter module still failed to start with the same error.

It was recommended by the OSG Technology Investigation group that I review two documents for possible information and hints on resolving these installation issues. The document on CVMFS by the University of Wisconsin-Madison [2] briefly mentions their server installation process, but in little detail, and mentions some problems that seemed orthogonal to this one: they had trouble getting data to sync in a particular set of circumstances, whereas we literally couldn't install the necessary packages. I confirmed with the University of Wisconsin-Madison document author that their local installation was done on RHEL 5. The OSG

document [3] was of little help, as it contained oblique references to a technology for serving repositories and lacked detail of any CVMFS deployment.

After multiple troubleshooting sessions and conversations with the CVMFS developers, I resigned to reverting to RHEL 5 for the server operating system. A new request was made to GCE for a new RHEL 5 VM, which was deployed within a few days; the previous RHEL 6 VM originally allocated for the server component was reconfigured as a CVMFS replica server. As with the CVMFS client on RHEL 6 before, after making a few package setting adjustments, the installation of a CVMFS server on RHEL 5 was relatively simple; however, as expected, the customization of the basic server mount point brought with it many complications.

The details of these issues are available upon request.

As a result of these findings, the customizations made to the current server were reverted, and the default settings and locations were used for testing server functions. To do so, the custom mount had to be destroyed, and two new ones created to accommodate the default CVMFS server mount locations:

- `/cvmfs` for source software storage, and
- `/srv` for CVMFS repository storage.

It was also necessary to install a Squid cache on the client system (ideally, this would be installed on its own host) that could interact with the replica server as a cache to the client.

Installation Details

The `iptables` firewall rules on all involved hosts must be modified in order to enable traffic between the server, replica, cache, and client.

RHEL 5 server install

1. Install `cvmfs-release` from remote URL [4].

   The `cvmfs-release` package is meant to automate installation of the CVMFS yum repository file and public signing key; however, the package is missing its own signing key and works only for Scientific Linux, not RHEL or other Enterprise Linux clones. As such, to install the package after downloading, override the default yum GPG key check:
   `yum --nogpgcheck install cvmfs-release-*.rpm`

   Afterward, replace the base URL in the yum repository file (`/etc/yum.repos.d/cernvm.repo`) with the functioning URL [4].

2. Ensure adequate space in two directories/partitions:
   - */cvmfs*
   - */srv*

3. Install the yum packages necessary for installing the CVMFS server:
```
yum install cvmfs-server cvmfs-keys httpd redirfs kmod-redirfs \
  cvmfsflt kmod-cvmfsflt
```

4. Test load the RedirFS and CVMFS kernel filter modules (no returned output indicates a successful test):
```
modprobe redirfs && modprobe cvmfsflt
```

5. After installation, the file `/etc/cvmfs/replica.conf`, which is an artifact of a package split, should be deleted.

CVMFS server and source repository configuration

1. Copy (`/etc/cvmfs/server.conf`) to a local override file (`/etc/cvmfs/server.local`) for customization.

   Note: this was done for testing purposes, which revealed that the desired modifications to source and shadow directories created a problematic instance. As a result, these customizations were rolled back to their defaults in the current instance.

2. Create a new CVMFS server "file system" and repository structure:
```
cvmfs_server mkfs {repository}
```

3. Touch the otherwise unused master replica file (undocumented bug fix):
```
touch /srv/cvmfs/{repository}/pub/catalogs/.cvmfs_master_replica
```

4. Synchronize data from the software source directory to the mirror tree:
```
cvmfs-sync
```

5. Publish the repository:
```
cvmfs_server publish
```

6. Copy the public signing key generated by the CVMFS server for the repository from the server to both the replica and the client:
```
/etc/cvmfs/keys/{repository}.pub
```

7. Start the CVMFS server daemon to monitor future updates:
```
service cvmfsd start
```

8. Script/cron regular repository syncing and publishing:

- */etc/cvmfs/cvmfs_publish.sh*

```
#!/bin/bash
chown -R cvmfs.cvmfs /cvmfs/testrepo.racf.bnl.gov
cvmfs-sync >>/var/log/cvmfs/update.log
cvmfs_server publish >>/var/log/cvmfs/update.log
```

- */etc/cron.d/cvmfs_publish*

```
*/20 * * * *  root /etc/cvmfs/cvmfs_publish.sh > /dev/null 2>&1
0 0 * * 0     root /usr/bin/cvmfs_server resign >>
/var/log/cvmfs/update.log 2>&1
```

RHEL 6 replica installation

1. Follow Step 1 of the server installation section to install the CVMFS repository file and public key.

2. Install the replica and Apache packages:
   `yum install cvmfs-replica cvmfs-keys httpd`

3. Add an entry to the CVMFS repository mirror file (`/etc/cvmfs/replica.repositories`) to point to the CVMFS server URL and repository path, signing key, and local path:
   `testrepo.racf.bnl.gov|http://grid24.usatlas.bnl.gov:80/cvmfs/testrepo.racf.bnl.gov|/etc/cvmfs/keys/testrepo.racf.bnl.gov.pub|/var/cvmfs/testrepo.racf.bnl.gov/pub|32|10|3`

4. Configure Apache to serve the repository replica:
   - */etc/httpd/conf.d/cvmfs-replica.conf*

```
RewriteEngine on
# /opt/<experiment> is forced to be lower case:
RewriteMap toLower int:tolower
# Automatically point to the catalogs:
RewriteRule ^/opt/([A-Za-z0-9\-\.]+)/(.*)$
/opt/${toLower:$1}/pub/catalogs/$2 [PT]
# Translation URL to real pathname:
Alias /opt /var/cvmfs

# Close access to non-pub directories
<DirectoryMatch /var/cvmfs/[A-Za-z0-9\-\.]+/(shadow|ctrl)>
  Order allow,deny
  Deny from all
</DirectoryMatch>

<Directory "/var/cvmfs">
  Options Indexes -MultiViews FollowSymLinks
  AllowOverride All
  Order allow,deny
  Allow from all
  IndexOptions +SuppressDescription +FoldersFirst +VersionSort
+SuppressIcon
  EnableMMAP Off
```

```
    EnableSendFile Off
    AddType application/x-cvmfs .cvmfschecksum .cvmfscatalog .x509 \
            .cvmfspublished .cvmfswhitelist
    FileETag INode MTime Size
    ExpiresActive On
    ExpiresDefault "access plus 3 days"
    ExpiresByType text/html "access plus 5 minutes"
    ExpiresByType application/x-cvmfs "access plus 15 minutes"
  </Directory>
```

5. Restart Apache:
   `service httpd restart`

6. Add a cron to update the local repository replica "snapshot" from the remote source:
   `20 * * * * root /usr/bin/cvmfs_snapshot > /dev/null 2>&1`

RHEL 6 cache installation

The ATLAS Frontier Squid package was installed to create a cache on the same machine as the test client. After installing the software via RPM [5], a single modification to the default settings was necessary to enable Squid to serve CVMFS content on the local network: in `/etc/squid/customize.sh`, edit the local access control list definition to include the local machine itself, and the network on which the replica and source server reside:

```
setoption("acl NET_LOCAL src", "127.0.0.1 130.199.0.0/16")
```

As a precaution, the default cache directory size was also increased from 10 GB to 40 GB:

```
setoptionparameter("cache_dir", 3, "40000")
```

The `frontier-squid` service was then restarted in order to enact these customizations.

RHEL 6 client installation

1. Follow Step 1 of the server installation section to install the CVMFS repository file and public key.

2. Install the client packages:
   `yum install cvmfs cvmfs-init-scripts cvmfs-keys`

3. Run the initial client configuration script:
   `cvmfs_config setup`

4. Create the following configuration files and example content to point to the new repository via the replica server (*not* the software source server):
   - */etc/cvmfs/default.local*
     ```
     CVMFS_REPOSITORIES=testrepo.racf.bnl.gov
     CVMFS_HTTP_PROXY="http://localhost:3128"
     ```
   - */etc/cvmfs/domain.d/{domain-of-repository}.conf*
     ```
     CVMFS_SERVER_URL=${CERNVM_SERVER_URL:="http://grid22.racf.bnl.gov/opt/@org@"}
     CVMFS_PUBLIC_KEY=/etc/cvmfs/keys/cern.ch.pub:/etc/cvmfs/keys/testrepo.racf.bnl.gov.pub:/etc/cvmfs/keys/cern-it2.cern.ch.pub:/etc/cvmfs/keys/cern-it3.cern.ch.pub
     ```
   - */etc/cvmfs/config.d/{repository}.conf*
     ```
     CVMFS_SERVER_URL=http://grid22.racf.bnl.gov/opt/testrepo.racf.bnl.gov
     ```
5. Restart CVMFS and `autofs`:
   ```
   service cvmfs restartautofs
   service cvmfs restartclean
   ```
6. Test the new client repository mount point:
   ```
   ls –la /cvmfs/{repository}
   ```

Confirmed Issues

According to developers, the current CVMFS server tools do not support main mount point renaming or customization of any kind; they were written exclusively to support the operations of creating and maintaining repositories in a controlled environment at CERN. A rewrite of the server mechanism is underway and may be available for test in a month.

Installing the CVMFS Release package cleanly requires a public signing key that is not included with the package itself. The package creates a yum repository file that is not platform-agnostic and requires a correction of the base URL in order to function.

All recommended CMVFS package fixes and improvements have been forwarded to the developers.

Current Test Environment

Test bed configuration:

- **grid24.usatlas.bnl.gov**: RHEL 5, CVMFS test repository server
- **grid22.racf.bnl.gov**: RHEL 6, CVMFS test repository replica
- **grid23.racf.bnl.gov**: RHEL 6, CVMFS test client *and* proxy/cache

Source location for installing a software repository:

- grid24.usatlas.bnl.gov:/cvmfs/testrepo.racf.bnl.gov/

Outstanding Issues

The software source needs to be populated with realistic content to enable proper testing of replica, cache, and client propagation.

Authentication and authorization mechanisms must be established on the server in order to protect the source content and enable code repository additions and updates.

References:

[1] https://rt.racf.bnl.gov/rt/Ticket/Display.html?id=22167
[2] http://www.hep.wisc.edu/cms/comp/ops/cvmfs.html
[3] http://tinyurl.com/docs-oasis
[4] http://cvmrepo.web.cern.ch/cvmrepo/yum/cvmfs/EL/5/x86_64/
[5] http://frontier.cern.ch/dist/rpms/